Amendments to the Drawings:

The attached replacement sheet of drawings includes changes to Fig. 3 and replaces the original sheet including Fig. 3.

In Figure 3, the legend "Prior Art" has been added.

Attachments following last page of this Amendment:

Replacement Sheet (1 page)

REMARKS

Claims 1-24 are currently pending. Claims 1, 5, 9, 14, 20, and 24 are independent. Claims 1 – 8, 14 – 16, 19 - 24 are currently amended. No new matter is added. Support for the amendments can be found at least at page 3, lines 3-14; page 7, lines 14-20; and page 8, lines 4-12 of the specification. Reconsideration of the action mailed May 16, 2007, is respectfully requested in view of the foregoing amendments and the following remarks.

The Examiner rejected claims 1-24 under obviousness-type double patenting as allegedly unpatentable over claims 1-24 of U.S. Patent No. 6,931,550 ("Rygaard"). The Examiner rejected claim 24 under 35 U.S.C. § 102(e) as allegedly anticipated by U.S. Patent No. 6,330,588 ("Freeman"). The Examiner rejected claims 1-23 under 35 U.S.C. § 103(a) as allegedly unpatentable over Jansen et al. "NIST Special Publication 800-19—Mobile Agent Security" ("Jansen") in view of U.S. Patent No. 6,233,601 ("Walsh") and Freeman. Applicant respectfully traverses the rejections.

Drawing Objections

The Examiner objected to FIG. 3 as requiring the legend "Prior Art". Applicant has amended FIG. 3 to include the legend "Prior Art".

The Examiner objected to Claim 4A for failing to include jumping application security system 50 and jumping application controller 64. Applicant has amended the specification, as shown above, to correct typographical errors. In particular, the specification discloses a server computer 52 (shown in FIG. 4A) which can be used as the jumping application security system. Additionally, jumping application controller 64 was corrected as jumping application controller module 140 as shown in FIG. 4A.

Applicant respectfully requests that the drawing objections be withdrawn.

Obviousness-type Double Patenting Rejection

Claims 1-24 stand rejected over claims 1-24 of Rygaard. The Examiner states that the claims of Rygaard are not patentably distinct from the claims of the present application. Applicant respectfully disagrees. The claims of Rygaard include features not found in the claims of the present application. Additionally, the claims of the present application include features not found in the claims of Rygaard.

For example, claim 1 of the present application includes a database that contains one or more pieces of code and a description of each piece of code, wherein each piece of code implements a particular behavior. The claims of Rygaard do not disclose such a database. Similarly, claim 1 of Rygaard discloses means for inspecting an access control list of a mobile application to determine if code is marked as immutable. The claims of the present application do not include this feature.

Therefore, Applicant submits that claims 1-24 are patentable distinct from the claims of Rygaard. Applicant respectfully requests that the double patenting rejection be withdrawn.

Section 102 Rejections

Claim 24 stands rejected over Freeman. Claim 24 is directed to a method that includes replacing code in the jumping application that implements a particular behavior with a piece of code that implements the particular behavior in the jumping application so that the jumping application has the particular behavior when it is executed by the second host for each jump of the jumping application from an untrusted host.

The Examiner states that Freeman discloses replacing code at col. 14, lines 7-18. Applicant respectfully disagrees. The cited portion of Freeman reads, in pertinent part, as follows:

> So implementing these mechanisms (and components) enables (a) the trusted resource to operate substantially protected from corruption and/or (b) the trusted resource's implicated mechanism(s) to be compared to, e.g., firmware copies from time to time so that, if any corruption is discovered, corrective measures can be invoked. Typical corrective measures include repairing software, e.g., by reloading objects, by deleting any extraneous code, by application of code comparison and correction algorithms or similar routines, and/or by other conventional approaches, whether automatically or by human initiation, such as by a system operator.

The cited portion of Freeman discloses using a resource to verify mobile software agents. The verification identifies and detects corruption in the mobile agent to protect the mobile agent.

The mobile agent can be corrected when corruption is identified. The correction includes repairing software by reloading an object, deleting code, and correcting code. However, Freeman does not disclose or suggest replacing code that implements a particular behavior with another piece of code that implements the same behavior when executed by a second host. Furthermore, Freeman does not disclose or suggest replacing the code for each jump of the jumping application from an untrusted host, as required by claim 24.

Applicant respectfully submits that claim 24 is in condition for allowance.

Section 103 Rejections

Claim 1 stands rejected over Jansen, Walsh, and Freeman. Claim 1 is directed to a jumping application security console that maintains the security of a jumping application that is jumping between two or more hosts connected to the security console. The security console includes a security module including instructions to replace code from the jumping application that implements a first behavior with a piece of code from the database into the jumping application that implements the first behavior for each jump of the jumping application from an untrusted host.

The Examiner acknowledges that neither Jansen nor Walsh disclose replacing code from a jumping application with a piece of code from a database. However, the Examiner states that Freeman does disclose replacing code at col. 13, lines 35-50 and col. 14, lines 7-18. Applicant respectfully disagrees. Col. 13, lines 35-50 read, in pertinent part, as follows:

> Toward accomplishing that end, the security mechanism typically is enabled to remove, disable or otherwise render ineffective any or all of a received agent that may be corrupted or corrupting. Such a mechanism preferably includes monitoring firmware and/or other hardware which is used to identify agents and other potentially executable code that may be corrupted.

The cited portion of col. 13 of Freeman discloses identifying and disabling corrupted code of a mobile agent. Similarly, col. 14, lines 7-18, as described with respect to claim 24 above, discloses using a resource to discover corruption in an agent and taking corrective measures. However, neither cited portion of Freeman discloses or suggests replacing code. Furthermore, the cited portions of Freeman disclose taking an action only when corrupted code is detected. However, claim 1 requires that particular code be replaced <u>for each jump</u> of the jumping application <u>from an untrusted host</u>. Freeman does not disclose or suggest a security console that replaces code from a jumping application each time jumping application jumps from an untrusted host.

Additionally, claim 1 requires that that code implement a particular behavior and that the replaced code implements the same behavior. Freeman does not disclose or suggest replacing code that implements a behavior with code that implements the same behavior when the jumping application jumps from an untrusted host.

Applicant respectfully submits that claim 1, as well as claims 2-4, which depend from claim 1, are in condition for allowance.

Claim 5 stands rejected over Jansen, Walsh, and Freeman. Claim 5 is directed to a jumping application security console maintains the security of a jumping application that is jumping between two or more hosts connected to the security console. The security console includes means for replacing code from the jumping application that implements a first behavior with a piece of code from the database into the jumping application that implements the first behavior during each jump between hosts. As set forth above with respect to claim 1, Jensen, Walsh, and Freeman fail to disclose or suggest replacing code in a jumping application when the jumping application jumps from an untrusted host.

Applicant respectfully submits that claim 5, as well as claims 6-8, which depend from claim 5, are in condition for allowance.

Claim 9 stands rejected over Jansen, Walsh, and Freeman. Claim 9 is directed to a method for controlling the security of a jumping application that includes replacing code in the jumping application that implements the particular behavior with a piece of code that implements the particular behavior in the jumping application so that the jumping application has the particular behavior when it is executed by a host in the jumping application system. As set forth above with respect to claim 1, Jensen, Walsh, and Freeman fail to disclose or suggest replacing code in a jumping application. Furthermore, the cited portions do not disclose or suggest receiving a request for a piece of code that implements a particular behavior for a jumping application.

Applicant respectfully submits that claim 9, as well as claims 10-13, which depend from claim 9, are in condition for allowance.

Claim 14 stands rejected over Jansen, Walsh, and Freeman. Claim 14 is directed to a jumping application security system that includes a security console having instructions that replace code from the jumping application that implements a first behavior with a piece of code from the database into the jumping application that implements the first behavior each time the jumping application jumps from an untrusted host. As set forth above with respect to claim 1, Jensen, Walsh, and Freeman fail to disclose or suggest replacing code in a jumping application each time jumping application jumps from an untrusted host.

Applicant respectfully submits that claim 14, as well as claims 15-19, which depend from claim 14, are in condition for allowance.

Claim 20 stands rejected over Jansen, Walsh, and Freeman. Claim 20 is directed to a server computer for a jumping application security system that includes replacing code from the jumping application that implements a first behavior with a piece of code from the database into the jumping application that implements the first behavior each time the jumping application jumps from a untrusted first host to a second host. As set forth above with respect to claim 1, Jensen, Walsh, and Freeman fail to disclose or suggest replacing code in a jumping application each time the jumping application jumps from an untrusted host.

Applicant respectfully submits that claim 20, as well as claims 21-23, which depend from claim 20, are in condition for allowance.

Conclusion

Applicant requests that all pending claims be allowed.

By responding in the forgoing remarks only to particular positions taken by the Examiner, Applicant does no acquiesce with other positions that have not been explicitly addressed. In addition, Applicants' arguments for patentability of a claim should not be understood as implying that no other reasons for the patentability of that claim exist. Finally, Applicant's decision to amend or cancel any claim should not be understood as implying that Applicant agrees with any positions taken by the Examiner with respect to that claim or other claims.

Please apply the required petition for extension of time fee of $120.00 and any other charges or credits to deposit account 06-1050.

Respectfully submitted,

Date: August 24, 2007

/Brian J. Gustafson/
Brian J. Gustafson
Reg. No. 52,978

PTO Customer No. 26181
Fish & Richardson P.C.
Telephone: (650) 839-5070
Facsimile: (650) 839-5071